10 ABSTRACT OF THE DISCLOSURE

The grammar of this invention is an approach to regular expressions which introduces advantages to programmers who use regular expressions for scanning, searching, and tokenizing text:

- Allowing composition of regular expressions (patterns) through the standard C/C++ operators (using C/C++ precedence and associativity), thus appealing to a huge audience of programmers already familiar with that style of grammar.
- Generalizing the negated character-class (so familiar to Perl programmers) in a way that quite dramatically allows ANY pattern composition to be used for searching (the "subjunctive" binary composition).
- Integrating into an elegantly simple grammar form ("do-pattern") the ability to create arbitrary side-effects of tokenization, accomplished in prior art through a cumbersome combination of tokenizing expressions and parse trees, such as in the grammar style of Lex-Yacc.
- Generalizing the capture-to-variable feature (as seen in Perl), allowing the capture of portions of the stream (match sub-expressions) into any variable current in the scope of the regular-expression.
- Allowing the parameterization of production rules, as templates, which allow similar (in form) regular expressions to be written as multiple instantiations of the same production rule template (accomplished via *in* params).
- Further allowing parameterization of production rule templates to extend to the capture output of the expressions (accomplished via a combination of "do-patterns", "capture-patterns", and *out* or *in/out* params).
- Creating novel support algorithms (to accomplish the above) not seen in any texts on finite automata.

The following table of contents is not intended for publication but is provided as an aid to prosecution of the present application since all page number references will change with publication.

Table of Contents

1 A NEW GRAMMAR FOR REGULAR EXPRESSIONS......1
2 CROSS-REFERENCES TO RELATED APPLICATIONS......1
3 STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT......1
4 REFERENCE TO SEQUENCE LISTING ON CD-ROM......1
5 BACKGROUND OF THE INVENTION......1
5.1 Field of the Invention......1
5.2 Description of Related Art......2
5.3 List of Prior Art......4
6 BRIEF SUMMARY OF INVENTION......5
6.1 Brief Description of Architecture......7
7 BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS......11
8 DETAILED DESCRIPTION OF INVENTION......12
8.1 Glossary of Terms......12
8.2 Coding Conventions and Standard Grammar Variations......16
8.3 Grammar Specification......18
8.3.1 Pattern Primitive......18
8.3.2 tokenize Statement......19
8.3.3 Stream data type......21
8.3.4 C-like grammar......22
8.3.5 Pattern Compositions......22
8.3.5.1 Irreducible (primary) expressions......27
8.3.5.2 Unary Pattern compositions......33
8.3.5.2.1 Unary Optional ?......35
8.3.5.2.2 Unary Repeat *......36
8.3.5.2.3 Unary Repeat +......38
8.3.5.3 Binary Pattern Compositions......39
8.3.5.3.1 Concatenation......40
8.3.5.3.2 Union......42
8.3.5.3.3 Repeat and Iterate......44
8.3.5.4 Pattern Compositions Based on Do-Patterns......50
8.3.5.4.1 Do-Pattern......50
8.3.5.4.2 Capture-Pattern......53
8.3.5.4.3 Production-Rule Grammar......55
8.3.5.5 Subjunctive Compositions......58
8.3.5.6 Other Compositions......61
8.3.5.6.1 String Grammar......61
8.3.5.6.2 Inhibit Grammar......62
8.3.5.7 Binding and Representing Expressions......62
8.4 Language Examples......64
8.4.1 Basic Examples......65
8.4.1.1 Literals and primary expressions......66

8.4.1.2 Unary operators....68
8.4.1.3 Binary operators....70
8.4.2 Special regular expressions....76
8.4.2.1 Capture-pattern....76
8.4.2.2 Do-Pattern Examples....83
8.4.2.3 Subjunctive Expressions....89
8.4.2.4 Production Rule Template Examples....100
8.4.2.5 Additional Examples....112
8.4.3 Advanced Examples....114
8.5 Automata Details....123
8.5.1 Virtual Machine....123
8.5.1.1 Abstract VM....127
8.5.1.2 FORTH VM....129
8.5.1.2.1 Standard Grammar....130
8.5.1.2.2 Pattern Expressions....134
8.5.1.2.3 Do-pattern Expressions....137
8.5.1.2.4 Tokenize Statement....138
8.5.1.2.5 Capture-pattern Expression....141
8.5.1.3 Production Rules....144
8.5.2 Automata Composition....154
8.5.2.1 Instruction Arcs....163
8.5.2.2 Arc Numbering....167
8.5.2.2.1 Ambiguity Sources....169
8.5.2.2.2 Sequences of Arcnum Sequences....172
8.5.2.2.3 Literals....173
8.5.2.2.4 Concatenation....173
8.5.2.2.5 Union....182
8.5.2.2.6 Repeats....184
8.5.2.2.7 Transformations and Short-circuits....191
8.5.2.2.8 Ambiguity Premise....195
8.5.3 Automata Construction....198
8.5.3.1 Transition Probing....201
8.5.3.2 Instruction Arcs....204
8.5.3.3 Arc Numbering....204
8.5.3.4 Subset Construction Rules....205
8.5.3.4.1 Definitions....205
8.5.3.4.2 Subset Construction Rules of the Invention....208
8.5.3.4.3 Premises....209
8.5.3.4.4 Justifications....211
8.5.4 Subjunctive Convolution....222
8.5.4.1 Overview....222
8.5.4.2 Union-Like....223
8.5.4.3 Transitions....223
8.5.4.4 Pruning....224
8.5.4.5 Exitability....225
8.5.4.6 Arrival-Time....227
8.5.4.7 Reachability....228
8.5.4.8 Re-composition....229

8.5.4.9 Example......229
8.5.5 Automata Execution......233
8.5.5.1 Example......236
8.6 Language Evolution......237
9 CLAIMS......242
10 ABSTRACT OF THE DISCLOSURE......255